

An ITK Filter for Bayesian Segmentation: itkBayesianClassifierImageFilter

John Melonakos¹, Karthik Krishnan² and Allen Tannenbaum¹

¹ Georgia Institute of Technology, Atlanta GA 30332, USA

{jmelonak, tannenba}@ece.gatech.edu,

² Kitware, Clifton Park NY 12065, USA

karthik.krishnan@kitware.com

January 9, 2006

Abstract. An Insight Toolkit (ITK) filter for image segmentation with applications to brain MRI scans is presented in this paper. Previously, we showed how ITK could be used to implement our algorithm. This paper presents our new ITK filter for Bayesian segmentation along with results on brain MRI scans. Our algorithm is a refinement of the work of Teo, Saprio, and Wandall. The basic idea is to incorporate prior knowledge into the segmentation through Bayes' rule. Image noise is removed via an affine invariant anisotropic smoothing of the posteriors as in Haker et. al. Specifically, we present the implementation of our Bayesian segmentation algorithm applied to brain MRI scans.

1 Introduction

In this paper, we present an Insight Toolkit (ITK) filter for image segmentation. Previously, we showed how ITK could be used to implement our algorithm [1]. This paper presents our new ITK filter for Bayesian segmentation along with results on brain MRI scans.

Our algorithm is a refinement of the work of Teo, Saprio, and Wandall [2]. The basic idea is to incorporate prior knowledge into the segmentation through Bayes' rule. Image noise is removed via an affine invariant anisotropic smoothing of the posteriors as in Haker et. al. [3]. Specifically, we present the implementation of our Bayesian segmentation algorithm applied to brain MRI scans.

This paper provides details about the inclusion of our Bayesian segmentation algorithm into ITK. In section 2, we provide a high-level overview of our algorithm. In section 3, we provide a User's Guide for our filter. In section 4, we demonstrate our filter on brain MRI scans. In section 5, we provide some concluding remarks.

We have submitted data and code in conjunction with this paper so that the reader may reproduce our results with ease. Section 4 will assist the user in reproducing our results.

2 Algorithm Details

In this section, we provide a high-level description of the Bayesian segmentation algorithm.

We assume that the value of each voxel intensity in a given class can be considered as a random variable, independent across pixels. In the following results, we assume that the voxel intensities are normally distributed. This assumption may be modified to support other distributions that may better fit the data. With a large set of training data, the distributions may also be learned a priori.

The application of the statistical distributions to the voxel intensities produces the membership probabilities, $Pr(Vi = v|Ci = c)$. The prior probabilities, $Pr(Ci = c)$, of a pixel belonging to a particular class are either provided by the user or assumed to be uniform.

Using the membership and prior probabilities, we generate the posterior probabilities via Bayes' Rule. Finally, using the maximum a posteriori (MAP) estimate on the posterior probabilities, the final labelmap is produced.

Teo et. al. and Haker et. al. have shown that smoothing the posteriors prior to applying the MAP estimate can often yield improved segmentations [2,3,4]. Our filter provides the user with the option to smooth the posteriors prior to applying the MAP estimate.

The following is a concise description of the algorithm:

Algorithm 1 Bayesian Segmentation High-Level Algorithm

Require: User specifies number of classes: N

- 1: Generate N images of membership probabilities by applying N user-defined statistical distributions to the raw data
 - 2: User provides N images of prior probabilities or, in the default case, uniform prior probabilities are assumed
 - 3: Generate N images of posterior probabilities via Bayes' Rule
 - 4: Smooth the posterior images for several iterations using an anisotropic PDE and renormalize the posterior images after each smoothing iteration
 - 5: Apply maximum a posteriori rule to achieve final segmented labelmap
-

3 User's Guide

In this section, we provide a User's Guide for this ITK filter. We proceed by describing the purpose of the accompanying code file by file.

3.1 itkBayesianClassifierInitializationImageFilter

itkBayesianClassifierInitializationImageFilter.(h/txx)

This filter is intended to be used as a helper class to initialize the BayesianClassifierImageFilter. The goal of this filter is to generate a membership image that indicates the membership of each pixel to each class. These membership images are fed as an input to the BayesianClassifierImageFilter.

Parameters Number of classes: This defines the number of classes, which will determine the number of membership images that will be generated. The user must specify this.

Membership functions: The user can optionally plugin in any membership function. The number of membership functions plugged in should be the same as the number of classes. If the user does not supply membership functions, the filter will generate membership functions for you. These functions are Gaussian density functions centered around N pixel intensity values, I_k . These N values are obtained by running K-means on the image. In other words, the default behavior of the filter is to generate a Gaussian mixture model for the input image.

Inputs and Outputs The filter takes a scalar image as input and generates a VectorImage, each component c of which represents memberships of each pixel to the class c .

Template parameters This filter is templated over the input image type and the data type used to represent the probabilities (defaults to float).

3.2 itkBayesianClassifierImageFilter

itkBayesianClassifierImageFilter.(h/txx)

Inputs and Outputs The input to this filter is an itk::VectorImage that represents pixel memberships to N classes. This image is conveniently generated by the BayesianClassifierInitializationImageFilter. You may use that filter to generate the membership images or specify your own.

The output of the filter is a label map (an image of unsigned char's is the default) with pixel values indicating the classes to which they correspond. Pixels with intensity 0 belong to the 0th class, 1 belong to the 1st class etc... The classification is done by applying a maximum decision rule to the posterior image.

Parameters The filter optionally allows you to specify a prior image. The prior image, if specified must be a VectorImage with as many components as the number of classes. The posterior image is then generated by multiplying the prior image with the membership image. If the prior image is not specified, the posterior image is the same as the membership image. Another way to look at it is that the priors default to having a uniform distribution over the number of classes. Posterior membership of a pixel = Prior * Membership.

The filter optionally accepts a smoothing filter and number of iterations associated with the smoothing filter. The philosophy is that the filter allows you to iteratively smooth the posteriors prior to applying the maximum decision rule. This offers a convenient approach to removing noise from imagery as in [2]. The user will need to plug in a smoothing filter with all the parameters set.

Template parameters InputVectorImage, datatype of the output labelmap, precision of the posterior image, precision of the prior image.

3.3 Examples for 1 and 2

BayesianClassifierInitializer.cxx BayesianClassifier.cxx
These files are examples for 1 and 2.

3.4 itkBayesianClassificationImageFilter

itkBayesianClassificationImageFilter.(h/txx)

This filter is a wrapper around the itkBayesianClassifierInitializationImageFilter and the itkBayesianClassifierImageFilter. It provides a minimal blackbox interface that takes an image (to be classified) as input and produces a label map.

3.5 itkBayesianClassificationImageFilterTest

itkBayesianClassificationImageFilterTest.cxx

Test for itkBayesianClassificationImageFilter. This is the code that we ran to produce the results described in the following section.

4 Results

In this section, we present results of this code on brain MRI scans. We also provide enough details to enable the reader to reproduce our results.

4.1 Overview

We present our results on brain MRI scans, comparing the results of the Bayesian segmentation to manual segmentations on datasets of schizophrenic patients. The patients' heads were imaged with a 1.5 T MRI system³ in the coronal plane and a postcontrast 3D sagittal spoiled gradient recalled (SPGR) acquisition with contiguous slices. The resolution is $0.975 \times 0.975 \times 1.5$ mm ($256 \times 256 \times 123$ voxels). All segmentations were done on 2D graylevel slices for simplicity in this publication, though the code can handle higher dimensional data. The Bayesian

³ Signa, GE Medical Systems, Milwaukee, WI

segmentations were obtained with the ITK code and data, which has been submitted in conjunction with this paper.

We compare the Bayesian segmentation (S) to the ground truth manual segmentation (G) using the DICE coefficient [5]: $DSC(S, G) := \frac{V_{S \cap G}}{\frac{1}{2}(V_S + V_G)}$, where V_X is the volume (number of voxels) of segmentation X . DSC values greater than 0.7 are regarded as good in the literature [5].

4.2 Details

Our results were obtained by running the `itkBayesianClassificationImageFilterTest` on 10 datasets of skull-removed imagery. For each case we picked the coronal slice immediately anterior to the temporal lobe tip. The `itkBayesianClassificationImageFilterTest` requires four command line parameters. For each dataset, we ran the code with the following parameters: `InputImageFile OutputImageFile 3 5`. We have included the raw data files along with the resulting segmentations for the reader to use in verifying the reproduction of this code.

We ran this code with ITK CVS at the time of this paper’s submission. The reader must use a version of ITK recent enough to properly compile and run this code.

4.3 DICE Results

The results (white matter mean DSC=0.8840 and gray matter mean DSC=0.8951 for $N = 10$ cases) show that the Bayesian segmenter gives good results in white matter and gray matter (see Table 1). The results of a typical Bayesian segmentation compared with the manual-based segmentations for Case 1 are shown in Figure 1(a),1(b) and for Case 2 in Figure 1(c),1(d).

	Case 1	Case 2	Case 3	Case 4	Case 5
Slice	96	99	101	104	98
WM DSC	0.8991	0.8586	0.8935	0.8830	0.8908
GM DSC	0.9051	0.8364	0.9121	0.8962	0.9072
	Case 6	Case 7	Case 8	Case 9	Case 10
Slice	101	98	97	98	98
WM DSC	0.8910	0.8635	0.9171	0.8877	0.8552
GM DSC	0.8921	0.8982	0.9278	0.8981	0.8779

Table 1. DICE validation measures for white (WM) and gray (GM) matter segmentations on 10 datasets

5 Conclusion

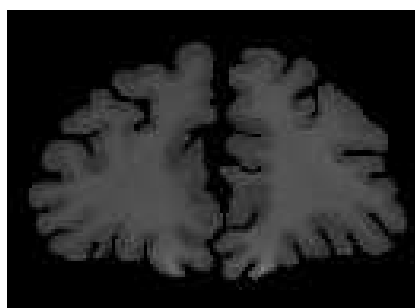
We have presented our ITK Bayesian segmentation code and shown positive results. User details and along with detailed explanations of our results provide the reader with the information necessary to reproduce these results. This code may also be found in the ITK CVS repository.

Acknowledgments

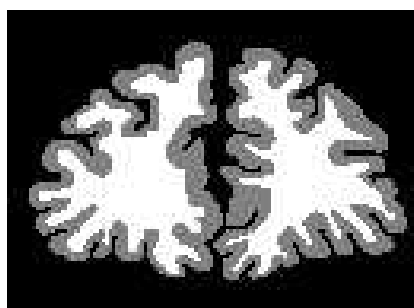
This work is part of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149. Information on the National Centers for Biomedical Computing can be obtained from <http://nihroadmap.nih.gov/bioinformatics>.

References

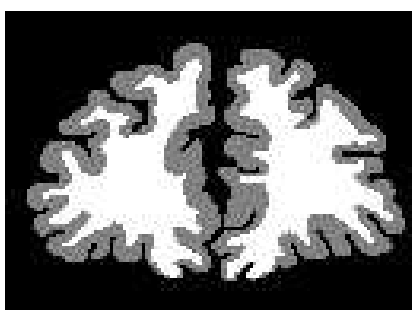
1. Melonakos, J., Al-hakim, R., Fallon, J., Tannenbaum, A.: Knowledge-based segmentation of brain mri scans using the insight toolkit. *Insight Journal* (2005) <http://hdl.handle.net/1926/44>.
2. Teo, P., Sapiro, G., Wandell, B.: Creating connected representations of cortical gray matter for functional MRI visualization. *IEEE Trans. Med. Imag.* **16** (1997) 852–863
3. Haker, S., Sapiro, G., Tannenbaum, A.: Knowledge-based segmentation of SAR data with learned priors. *IEEE Trans. Image Proc.* **9** (2000) 298–302
4. Teo, P., Sapiro, G., Wandell, B.: Anisotropic smoothing of posterior probabilities. In: *In Proc. ICIP, Santa Barbara, CA.* (1997)
5. Zijdenbos, A., Dawant, B., Marjolin, R.: Morphometric analysis of white matter lesions in mr images: Methods and validation. *IEEE TMI* **13** (1994) 716–724



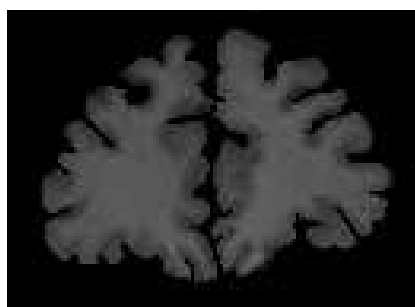
(a) Raw (Case 1)



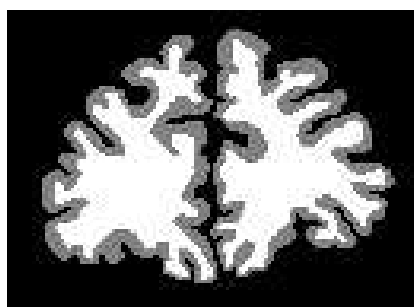
(b) Manual (Case 1)



(c) Bayesian (Case 1)



(d) Raw (Case 2)



(e) Manual (Case 2)



(f) Bayesian (Case 2)

Fig. 1.